

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1832	(space with allocat\$3) and (free with block)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:40
L2	40	waf1 and 1	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:27
L3	0	2 and @ad<"2000818"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:28
L4	0	1 and @ad<"2000818"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:27
L5	6	2 and @ad<"20000818"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:31
L6	1	"688070".apn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:30
L7	18275	(block with binary)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:30
L8	171	1 and 7	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:30

EAST Search History

L9	576	writ\$3 with (block with binary)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:31
L10	5	1 and 9	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:31
L11	70169	writ\$3 with (block)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:40
L12	123	8 and 11	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:31
L13	47	12 and @ad<"20000818"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:41
L14	1	13 and (snapshot with block\$2)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:32
L15	85	(space with allocat\$3) and wafi	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:40
L16	68	11 and 15	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:41

EAST Search History

L17	6	16 and @ad<"20000818"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:19
L18	32	1 and (707/202-204).ccls. and @ad<"20000818"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:11
L19	23	18 and 11	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:18
L20	336	1 and (block\$2 with region)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:18
L21	1520	(writ\$3 with block with region)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:19
L22	69	20 and 21	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:19
L23	26	22 and @ad<"20000818"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:20
L24	1073	(block with snapshot)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:19

EAST Search History

L25	0	23 and 24	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:20
L26	10	20 and 24	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:20
L27	0	26 and @ad<"20000818"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 13:20
S1	2	("6636879").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/07/25 12:20
S2	1	"6119244".PN.	USPAT; USOCR	OR	OFF	2007/07/25 10:09
S3	1	"6000039".PN.	USPAT; USOCR	OR	OFF	2007/07/25 10:09

[Web](#) [Images](#) [Video](#) [News](#) [Maps](#) [Gmail](#) [more ▾](#)[Sign in](#)

Google

region block waf snapshot space allocation

Search

[Advanced Search](#)
[Preferences](#)

Web

Results 1 - 10 of about 111 for **region block waf snapshot space allocation**. (0.42 seconds)**Space allocation in a write anywhere file system - Patent 20040139273**

[0007] One aspect of a **WAFL** (Write Anywhere File Layout) file system ... 2 shows a **block** diagram of a system for improved **space allocation** on disk storage. ...

www.freepatentsonline.com/20040139273.html - 48k - [Cached](#) - [Similar pages](#)

Instant snapshot - Patent 20020083037

Snapshot block ownership in **WAFL** has been recorded by updating the vacant blocks in a **region** containing a fixed number of blocks, such as 1024 blocks. ...

www.freepatentsonline.com/20020083037.html - 57k - Supplemental Result -

[Cached](#) - [Similar pages](#)

[[More results from www.freepatentsonline.com](#)]

File system defragmentation technique via write allocation - US ...

A free **space block** represents an available location (e.g., data **block** 404) on ... at the consistency point, the **WAFL** file system saves it as a **snapshot**. ...

www.patentstorm.us/patents/6978283-description.html - 69k - [Cached](#) - [Similar pages](#)

[PDF] Disk Performance of Copy-On-Write Snapshot Logical Volumes

File Format: PDF/Adobe Acrobat - [View as HTML](#)

on our findings about the performance of the LVM2 **snapshot** volumes, we propose. new disk **space allocation** and **block** placement techniques and build a ...

www.cs.ubc.ca/grads/resources/thesis/Nov06/Bhavana.pdf - [Similar pages](#)

Instant snapshot - Patent Review 7072916

The write **allocation** code decides whether a **block** is free by examining the active map and For **WAFL** to find free **space** to write new data or metadata, ...

www.wikipatents.com/7072916.html - 283k - [Cached](#) - [Similar pages](#)

[PDF] SUPPORTING SNAPSHOTS IN A LOG BASED FILE SYSTEM By ATAUR RAHIM ...

File Format: PDF/Adobe Acrobat - [View as HTML](#)

the **block** is set; so the **block** is available for **allocation**. At time T2, **WAFL** allocates the. **block** and stores the data in it. At time T3 and T4, **snapshot** #1 ...

purl.fcla.edu/fcla/etd/UFE0008900 - [Similar pages](#)

[PDF] Proceedings of the FAST 2002 Conference on File and Storage ...

File Format: PDF/Adobe Acrobat

saved fsinfo **block**. **WAFL** maintains the **block** allocations for each. **snapshot** in its own active map file. The active map file. is an array with one **allocation** ...

www.usenix.org/events/fast02/full_papers/patterson/patterson.pdf - [Similar pages](#)

[PS] A Persistent Snapshot Device Driver for Linux

File Format: Adobe PostScript - [View as Text](#)

disk **space**. Before taking a **snapshot**, the le sys-. tem or raw partition has to be frozen. alternatives is to divide the **block allocation** area, ...

www.sagecertification.org/publications/

[library/proceedings/als01/full_papers/siddha/siddha.ps](#) - [Similar pages](#)

[PDF] UNIVERSITY OF CALIFORNIA SANTA CRUZ STORAGE MANAGEMENT IN LARGE ...

File Format: PDF/Adobe Acrobat - [View as HTML](#)

ject **allocation** involves two steps: **region allocation** and **block allocation**. Rather than pre-allocating fixed disk **space** for the. metadata, **WAFL** uses ...
www.ssrc.ucsc.edu/Papers/wang-phdthesis06.pdf - [Similar pages](#)

[PDF] [Ext3cow: The Design, Implementation, and Analysis of Metadata for ...](#)

File Format: PDF/Adobe Acrobat - [View as HTML](#)

To examine the effect of **snapshot** on metadata **allocation**, we used three months of the same **block**. Ext3cow will read this small **region** in a single I/O, ...
hssl.cs.jhu.edu/papers/peterson-ext3cow03.pdf - [Similar pages](#)

[1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) **Next**

Download [Google Pack](#): free essential software for your PC

[Search within results](#) | [Language Tools](#) | [Search Tips](#) | [Dissatisfied?](#) [Help us improve](#)

©2007 Google - [Google Home](#) - [Advertising Programs](#) - [Business Solutions](#) - [About Google](#)

**Search Results**[BROWSE](#)[SEARCH](#)[IEEE XPLORE GUIDE](#)

Results for "(region block waf snapshot space allocation<in>metadata)"

e-mail

Your search matched 0 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by **Relevance** in **Descending** order.

» Search Options

[View Session History](#)[New Search](#)**Modify Search****Search**☐ Check to search only within this results setDisplay Format: ☒ Citation ☐ Citation & Abstract

» Key



Indicates full text access

IEEE JNL IEEE Journal or Magazine

IET JNL IET Journal or Magazine

IEEE CNF IEEE Conference Proceeding

IET CNF IET Conference Proceeding

IEEE STD IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistance.

[Help](#) [Contact Us](#) [Privacy & :](#)

© Copyright 2006 IEEE –

Indexed by
 Inspec®


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

 Terms used: **region block wafI snapshot space allocation**

 Found **45,712** of **207,474**

Sort results by


[Save results to a Binder](#)
[Try an Advanced Search](#)
[Try this search in The ACM Guide](#)

Display results


[Search Tips](#)
☐ Open results in a new window

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

 Relevance scale ☐ ☐ ☐ ☐ ☐

1 [An approach to virtual allocation in storage systems](#)



Sukwoo Kang, A. L. Narasimha Reddy

 November 2006 **ACM Transactions on Storage (TOS)**, Volume 2 Issue 4

Publisher: ACM Press

 Full text available: pdf(960.24 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This article presents *virtual allocation*, a scheme for flexible storage allocation. Virtual allocation separates storage allocation from the file system. It employs an allocate-on-write strategy which lets applications fit into the actual usage of storage space, without regard to the configured file system size. This improves flexibility by allowing storage space to be shared across different file systems. This article presents the design of virtual allocation and its evaluation through b ...

Keywords: Storage systems, file systems, storage allocation, storage management

2 [The Conquest file system: Better performance through a disk/persistent-RAM hybrid design](#)



An-I Andy Wang, Geoff Kuenning, Peter Reiher, Gerald Popek

 August 2006 **ACM Transactions on Storage (TOS)**, Volume 2 Issue 3

Publisher: ACM Press

 Full text available: pdf(1.34 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Modern file systems assume the use of disk, a system-wide performance bottleneck for over a decade. Current disk caching and RAM file systems either impose high overhead to access memory content or fail to provide mechanisms to achieve data persistence across reboots. The *Conquest* file system is based on the observation that memory is becoming inexpensive, which enables all file system services to be delivered from memory, except for providing large storage capacity. Unlike caching, *Con* ...

Keywords: Persistent RAM, file systems, performance measurement, storage management

3 [Versatility and Unix semantics in namespace unification](#)



Charles P. Wright, Jay Dave, Puja Gupta, Harikesavan Krishnan, David P. Quigley, Erez Zadok, Mohammad Nayyer Zubair

 February 2006 **ACM Transactions on Storage (TOS)**, Volume 2 Issue 1

Publisher: ACM PressFull text available:  pdf(317.82 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Administrators often prefer to keep related sets of files in different locations or media, as it is easier to maintain them separately. Users, however, prefer to see all files in one location for convenience. One solution that accommodates both needs is virtual namespace unification---providing a merged view of several directories without physically merging them. For example, namespace unification can merge the contents of several CD-ROM images without unpacking them, merge binary directories fr ...

Keywords: Namespace management, directory merging, snapshotting, stackable file systems, unification

4 [A region-based memory manager for prolog](#)



Henning Makhholm

October 2000 **ACM SIGPLAN Notices , Proceedings of the 2nd international symposium on Memory management ISMM '00**, Volume 36 Issue 1**Publisher:** ACM PressFull text available:  pdf(1.34 MB) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

We extend Tofte and Talpin's region-based model for memory management to support backtracking and cuts, which makes it suitable for use with Prolog and other logic programming languages. We describe how the extended model can be implemented and report on the performance of a prototype implementation. The prototype implementation performs well when compared to a garbage-collecting Prolog implementation using comparable technology for non-memory-management issues.

5 [Distributed FIFO allocation of identical resources using small shared space](#)



Michael J. Fischer, Nancy A. Lynch, James E. Burns, Allan Borodin

January 1989 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 11 Issue 1**Publisher:** ACM PressFull text available:  pdf(1.80 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

We present a simple and efficient algorithm for the FIFO allocation of k identical resources among asynchronous processes that communicate via shared memory. The algorithm simulates a shared queue but uses exponentially fewer shared memory values, resulting in practical savings of time and space as well as program complexity. The algorithm is robust against process failure through unannounced stopping, making it attractive also for use in an environment of processes of wide ...

6 [On bounding time and space for multiprocessor garbage collection](#)



Guy E. Blelloch, Perry Cheng

May 1999 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1999 conference on Programming language design and implementation PLDI '99**, Volume 34 Issue 5**Publisher:** ACM PressFull text available:  pdf(1.85 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper presents the first multiprocessor garbage collection algorithm with provable bounds on time and space. The algorithm is a real-time shared-memory copying collector. We prove that the algorithm requires at most $2(R(1 + 2/k) + N + 5PD)$ memory locations,

where P is the number of processors, R is the maximum reachable space during a computation (number of locations accessible from the root set), N is the maximum number of reachable objects, ...

7 Measurement: The impact of address allocation and routing on the structure and implementation of routing tables



Harsha Narayan, Ramesh Govindan, George Varghese

August 2003 **Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications SIGCOMM '03**

Publisher: ACM Press

Full text available: [pdf\(148.92 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The recent growth in the size of the routing table has led to an interest in quantitatively understanding both the causes (eg multihoming) as well as the effects (eg impact on router lookup implementations) of such routing table growth. In this paper, we describe a new model called **ARAM** that defines the structure of routing tables of any given size. Unlike simpler empirical models that work backwards from effects (eg current prefix length distributions), **ARAM** a ...

Keywords: IP lookups, modeling, routing tables

8 Comparison of access methods for time-evolving data



Betty Salzberg, Vassilis J. Tsotras

June 1999 **ACM Computing Surveys (CSUR)**, Volume 31 Issue 2

Publisher: ACM Press

Full text available: [pdf\(529.53 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper compares different indexing techniques proposed for supporting efficient access to temporal data. The comparison is based on a collection of important performance criteria, including the space consumed, update processing, and query time for representative queries. The comparison is based on worst-case analysis, hence no assumptions on data distribution or query frequencies are made. When a number of methods have the same asymptotic worst-case behavior, features in the methods tha ...

Keywords: I/O performance, access methods, structures, temporal databases

9 Real-time shading



Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**

Publisher: ACM Press

Full text available: [pdf\(7.39 MB\)](#) Additional Information: [full citation](#), [abstract](#)

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...

10 Fusion-based register allocation

Guei-Yuan Lueh, Thomas Gross, Ali-Reza Adl-Tabatabai

May 2000 **ACM Transactions on Programming Languages and Systems (TOPLAS)**,

Volume 22 Issue 3

Publisher: ACM PressFull text available: pdf(475.45 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

The register allocation phase of a compiler maps live ranges of a program to registers. If there are more candidates than there are physical registers, the register allocator must spill a live range (the home location is in memory) or split a live range (the live range occupies multiple locations). One of the challenges for a register allocator is to deal with spilling and splitting together. Fusion-based register allocation uses the structure of the program to make splitting and spilling d ...

Keywords: performance evaluation, register allocation**11** Smalltalk-80: the language and its implementation

Adele Goldberg, David Robson

January 1983 Book

Publisher: Addison-Wesley Longman Publishing Co., Inc.Full text available: pdf(33.56 MB) Additional Information: [full citation](#), [abstract](#), [cited by](#), [index terms](#), [review](#)**From the Preface (See Front Matter for full Preface)**

Advances in the design and production of computer hardware have brought many more people into direct contact with computers. Similar advances in the design and production of computer software are required in order that this increased contact be as rewarding as possible. The Smalltalk-80 system is a result of a decade of research into creating computer software that is appropriate for producing highly functional and interactive ...

12 Frangipani: a scalable distributed file system

Chandramohan A. Thekkath, Timothy Mann, Edward K. Lee

October 1997 **ACM SIGOPS Operating Systems Review , Proceedings of the sixteenth ACM symposium on Operating systems principles SOSP '97**, Volume 31 Issue 5**Publisher:** ACM PressFull text available: pdf(2.20 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)**13** Compiler construction: an advanced courseF. L. Bauer, F. L. De Remer, M. Griffiths, U. Hill, J. J. Horning, C. H. A. Koster, W. M. McKeeman, P. C. Poole, W. M. Waite, G. Goos, J. Hartmanis
January 1974 Book**Publisher:** Springer-Verlag New York, Inc.Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#)

The Advanced Course took place from March 4 to 15, 1974 and was organized by the Mathematical Institute of the Technical University of Munich and the Leibniz Computing Center of the Bavarian Academy of Sciences, in co-operation with the European Communities, sponsored by the Ministry for Research and Technology of the Federal Republic of Germany and by the European Research Office, London.

14 Essays in computing science

C. A. R. Hoare

January 1989 Book

Publisher: Prentice-Hall, Inc.

Full text available:  pdf(20.91 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [review](#)

Charles Antony Richard Hoare is one of the most productive and prolific computer scientists. This volume contains a selection of his published papers. There is a need, as in a Shakespearian Chorus, to offer some apology for what the book manifestly fails to achieve. It is not a complete 'collected works'. Selection between papers of this quality is not easy and, given the book's already considerable size, some difficult decisions as to what to omit have had to be made. Pity the editor weighin ...



15 [Exploiting perception in high-fidelity virtual environments: Exploiting perception in high-fidelity virtual environments](#)



Additional presentations from the 24th course are available on the citation page

Mashhuda Glencross, Alan G. Chalmers, Ming C. Lin, Miguel A. Otaduy, Diego Gutierrez
July 2006 **ACM SIGGRAPH 2006 Courses SIGGRAPH '06**

Publisher: ACM Press

Full text available:  pdf(5.07 MB)  mov(68:6 MIN) Additional Information: [full citation](#), [appendices and supplements](#), [abstract](#), [references](#), [cited by](#), [index terms](#)

The objective of this course is to provide an introduction to the issues that must be considered when building high-fidelity 3D engaging shared virtual environments. The principles of human perception guide important development of algorithms and techniques in collaboration, graphical, auditory, and haptic rendering. We aim to show how human perception is exploited to achieve realism in high fidelity environments within the constraints of available finite computational resources. In this course w ...

Keywords: collaborative environments, haptics, high-fidelity rendering, human-computer interaction, multi-user, networked applications, perception, virtual reality

16 [GPGPU: general purpose computation on graphics hardware](#)



David Luebke, Mark Harris, Jens Krüger, Tim Purcell, Naga Govindaraju, Ian Buck, Cliff Woolley, Aaron Lefohn

August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**

Publisher: ACM Press

Full text available:  pdf(63.03 MB) Additional Information: [full citation](#), [abstract](#), [citations](#)

The graphics processor (GPU) on today's commodity video cards has evolved into an extremely powerful and flexible processor. The latest graphics architectures provide tremendous memory bandwidth and computational horsepower, with fully programmable vertex and pixel processing units that support vector operations up to full IEEE floating point precision. High level languages have emerged for graphics hardware, making this computational power accessible. Architecturally, GPUs are highly parallel s ...

17 [Visualising the train garbage collector](#)



Tony Printezis, Alex Garthwaite

June 2002 **ACM SIGPLAN Notices , Proceedings of the 3rd international symposium on Memory management ISMM '02**, Volume 38 Issue 2 supplement

Publisher: ACM Press

Full text available:  pdf(332.90 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper presents a novel method for visualising an incremental garbage collector, based on the well-known Train algorithm, that generates concise snapshots of its state and informative graphs of its operation over time. To obtain these visualisations we used

GC-spy, a generic heap visualisation framework. We show how this easy-to-use tool provided a visualisation model that was effective in both confirming our pre-existing beliefs about the collector's operation and, more interestingly, highl ...

18 Ext3cow: a time-shifting file system for regulatory compliance



Zachary Peterson, Randal Burns

May 2005 **ACM Transactions on Storage (TOS)**, Volume 1 Issue 2

Publisher: ACM Press

Full text available: pdf(443.01 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The ext3cow file system, built on the popular ext3 file system, provides an open-source file versioning and snapshot platform for compliance with the versioning and auditability requirements of recent electronic record retention legislation. Ext3cow provides a *time-shifting* interface that permits a real-time and continuous view of data in the past. Time-shifting does not pollute the file system namespace nor require snapshots to be mounted as a separate file system. Further, ext3cow is i ...

Keywords: Versioning file systems, copy-on-write

19 Join operations in temporal databases



Dengfeng Gao, S. Jensen, T. Snodgrass, D. Soo

March 2005 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 14 Issue 1

Publisher: Springer-Verlag New York, Inc.

Full text available: pdf(374.28 KB) Additional Information: [full citation](#), [abstract](#), [citations](#)

Joins are arguably the most important relational operators. Poor implementations are tantamount to computing the Cartesian product of the input relations. In a temporal database, the problem is more acute for two reasons. First, conventional techniques are designed for the evaluation of joins with equality predicates rather than the inequality predicates prevalent in valid-time queries. Second, the presence of temporally varying data dramatically increases the size of a database. These factors i ...

Keywords: Attribute skew, Interval join, Partition join, Sort-merge join, Temporal Cartesian product, Temporal join, Timestamp skew

20 GCspy: an adaptable heap visualisation framework



Tony Printezis, Richard Jones

November 2002 **ACM SIGPLAN Notices , Proceedings of the 17th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications OOPSLA '02**, Volume 37 Issue 11

Publisher: ACM Press

Full text available: pdf(215.66 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

GCspy is an architectural framework for the collection, transmission, storage and replay of memory management behaviour. It makes new contributions to the understanding of the dynamic memory behaviour of programming languages (and especially object-oriented languages that make heavy demands on the performance of memory managers). GCspy's architecture allows easy incorporation into *any* memory management system: it is not limited to garbage-collected languages. It requires only small change ...

Keywords: Java, garbage collection, language implementation, memory management, visualisation of objects

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2007 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)